L33: Entry 165 of 175 File: USPT Aug 15, 2000

DOCUMENT-IDENTIFIER: US 6105008 A
TITLE: Internet loading system using smart card

Brief Summary Text (4):
With the explosive growth in open networks (such as the Internet) over the past several years and the rapid increase in the number of consumers with access to the World Wide Web, there has been a great deal of interest in the development of electronic commerce on the Internet. Traditional financial transactions are being transformed.

Brief Summary Text (5):
A technique for performing financial transactions uses a smart card. A smart card is typically a credit card-sized plastic card that includes a semiconductor chip for holding the digital equivalent of cash directly, instead of pointing to an account or providing credits. One example of a smart card is illustrated in FIG. 1. Of course, a smart card may be implemented in many ways, and need not necessarily include a microprocessor or other features. The smart card may be programmed with various types of functionality, such as a stored-value application; credit/debit; loyalty programs, etc. For the purpose of this disclosure, card 5 is programmed at least with a stored-value application, and will be referred to as "stored-value" card 5.

Brief Summary Text (7):
Microprocessor 12 is any suitable central processing unit for executing commands and controlling the device. RAM 14 serves as storage for calculated results and as stack memory. ROM 16 stores the operating system, fixed data, standard routines, and look up tables. Non-volatile memory 18 (such as EPROM or EEPROM) serves to store information that must not be lost when the card is disconnected from a power source but that must also be alterable to accommodate data specific to individual cards or any changes possible over the card lifetime. This information might include a card identification number, a personal identification number, authorization levels, cash balances, credit limits, etc. Encryption module 22 is an optional hardware module used for performing a variety of encryption algorithms. Card reader interface 24 includes the software and hardware necessary for communication with the outside world. A wide variety of interfaces are possible. By way of example, interface 24 may provide a contact interface, a close-coupled interface, a remote-coupled interface, or a variety of other interfaces. With a contact interface, signals from the microcontroller are routed to a number of metal contacts on the outside of the card which come in physical contact with similar contacts of a card reader device.

Brief Summary Text (25):
Furthermore, a bank need only make a minimal investment in time and money to take advantage of the present invention in order to allow its customers to load value from their existing accounts over the Internet. The bank need not engage in the development of complex custom software or accounting procedures. By incorporating software libraries, a bank is ready to begin loading value onto its customer's cards from its web site. Preferably, libraries are provided that interface with an existing server at a bank to facilitate the building of an HTML page. Because a smart card with a stored-value application is used, the bank server, load server

and client terminal perform the details of the transaction and the bank itself is relieved from having to control and keep track of a transaction. Also, the load server and stored-value card manage and provide security for the transaction. I.e., the bank need not be concerned about security nor be responsible for authenticating a stored-value card nor for determining a balance on the card. Of course, a load server could coexist alongside the bank server or could even be the same computer. That is, a bank could implement load server functionality at its own site if it so desired. In a preferred embodiment, the load server and its security module is provided by a separate financial institution or by a third-party processor.

Brief Summary Text (27):
The present invention is suitable for use with any type of stored-value card that is able to store an amount and to load a value upon a command. In one embodiment of the invention, a stored-value card implemented as a processor card works well. Use of a processor card has advantages where information processing is done on the card rather than in the terminal or host computer. Processor cards allow encryption to be done by the card, allow generation of signatures, and can accommodate multiple passwords or personal identification (such as biometrics that uniquely identify the holder of the card). Processor cards also provide increased data security, an anti-fraud capability, flexibility in applications, a multi-purpose capability, and off-line validation. Because high telecommunication costs and/or low reliability of a network may make on-line authorization impractical, a stored-value card with the capability for performing off-line processing and authentication by itself is extremely valuable.

Drawing Description Text (5):
FIG. 3 is a block diagram of an example of a clearing and administration system useful for reconciling financial transactions received from a service payment terminal.

Drawing Description Text (18):
FIG. 16 illustrates an architecture and system for authentication over an internet using a stored-value card.

Detailed Description Text (6):
A stored-value card may also perform a variety of functions in addition to simply storing value. A card may be dedicated to the storing value or may contain memory and programs for other applications as well. By way of example, an "electronic wallet" refers to a processor card that can execute a variety of financial transactions and identification functions. Such a card may serve debit, credit, prepayment, and other functions. A stored-value card typically includes information such as a bank identifier number, a sequence number, a purchase key, a load key, an update key, an expiration date, a transaction counter, a session key, etc., in addition to a running balance.

Detailed Description Text (17):
By way of example, security card 218 is a removable credit card-sized processor card that is programmed to process and store data relating to financial transactions. Security card 218 contains a microchip embedded in the card that enables the security card to authenticate and to validate the user's stored-value card. If a user stored-value card is accepted by the security card, and the stored-value card contains sufficient value, the security card guarantees that the merchant providing the goods and/or services receives payment according to the amount deducted from the stored-value card for the goods and/or services rendered. In a preferred embodiment, the security card also contains DES purchase security keys and authenticates the stored-value card during a purchase transaction and secures the payment and collection totals. A security card also stores signature algorithms for stored-value cards in use. A security card may also contain a transaction identifier for the current transaction, a financial sum of all transactions remaining to be settled, a session key, and master keys for all

stored-value cards in use. Further, the security card may contain generations of keys, blocked card indicators, date of last update, multiple card programs, different currency rates and additional security.

Detailed Description Text (20):
A brief discussion of the flow of a transaction now follows. During a financial transaction, the client terminal and merchant server exchange information 234 via internet 202. Each transaction initiated by a user has a transaction identifier created at the merchant server, and a merchant identifier unique to the payment server is also available from the merchant server. Client module 224 and the payment server also use this unique transaction identifier for tracking and logging information about the transaction. Merchant server 208 generates a unique identification of the transaction, completes other required parameters, encrypts as appropriate, and builds an HTML page and sends it to the client terminal. The client module interacts 235 with the stored-value card and builds a draw request message containing related card information, the purchase amount, and other information supplied by the merchant server.

Detailed Description Text (26):
FIG. 5 illustrates a detailed embodiment of internet payment architecture 200 having client terminal 204, payment server 206 and merchant server 208. A stored-value card 5 is in communication with client terminal 204, and a security card 218 inside a terminal 214 is in communication with payment server 206. Not shown for simplicity in this figure are other elements of the system shown in FIG. 4. One embodiment of a technique by which a financial transaction may be completed over the Internet will now be described using the flowchart of FIGS. 11A through 11D with reference to FIG. 5.

Detailed Description Text (33):
To operate securely and reliably in this environment, in one embodiment of the present invention, client module 224 emulates a security card and gathers all the responses for transmission in one draw request message. The draw request message may include a variety of information including a draw request token, state information, the merchant identifier, the transaction identifier, security information, a purse provider identifier, an intersector electronic purse (IEP) identifier, an algorithm used by the card, an expiry date, the balance of the card, a currency code, a currency exponent, the authentication mode of the IEP, the transaction number of the IEP, a key version and the purchase amount. As all of this information is prepackaged into a single draw request message, the number of messages between the stored-value card and the security card over the Internet is greatly reduced.

Detailed Description Text (46):
In step 636 the client terminal then passes this confirmation message 330 on to the merchant server at the URL address previously received from the merchant server. Message 330 may also be termed a "message result." The client may also post a message to the user informing that the debit has been completed. The client also logs confirmation of the payment. In step 638 the merchant server registers this confirmation message and checks for success. The merchant server calls a validate routine within the merchant code module with the confirmation message in order to validate the response from the client. The validate routine is able to take the transaction identifier along with the encrypted confirmation message to decrypt the confirmation message. If the decrypted confirmation message is acceptable, the merchant server then determines a successful transaction has occurred. Next, in step 640 the merchant server generates an HTML page with the purchased information and delivers this information to the client terminal. Alternatively, the merchant server may generate a purchase receipt to deliver to the client terminal indicating goods and/or services to be rendered. At this point, the client terminal may also log the merchant server's response. Completion of these steps indicates a successful financial transaction over the Internet using a stored-value card.

Detailed Description Text (87):
AUTHENTICATION EMBODIMENT

Detailed Description Text (88):
FIG. 16 illustrates an architecture and system 200' for authentication over an internet (such as the Internet) using a pseudo stored-value application. This application could reside on a stored-value card along with standard accounts, stored value, or other card applications. The card defines access to the pseudo stored-value service and ensures that the card is present and passes security checks.

Detailed Description Text (89):
In one embodiment of the present invention, a consumer may wish to access any of a variety of Web servers in order to redeem frequent flyer miles, award points, etc., that he or she has accumulated. In this embodiment, a consumer has accumulated "points" through any of a variety of programs with airlines, restaurants, rental car companies, hotels, banks, credit or debit card issuers, telephone or other communication company, etc. The consumer wishes to redeem these points to receive free airline tickets, meals, car rental, overnight stays, prizes, awards, discounts, or other "benefits". By accessing a Web server associated with the particular program, the consumer is able to use his or her card in any of the embodiments described herein to authenticate the card and to receive these benefits from the program. Most often, a card has a card number that is associated with the consumer's name in a database on the Web server. This card number is transmitted to the Web server as part of the card signature, or in a similar fashion. Thus, an authenticated card used in this embodiment to redeem services may be matched to the appropriate consumer.

Detailed Description Text (90):
For example, a consumer with 30,000 frequent flyer miles on one airline may use this embodiment of the present invention to access a Web server associated with the airline. The consumer is requesting a free round-trip ticket in exchange for 20,000 miles. The present invention then operates to authenticate the consumer's stored-value loyalty application on the card, and delivers a confirmation of authentication message to the Web server for the airline. The Web server then deducts 20,000 miles from the consumer's account (leaving 10,000 miles) and delivers the free ticket to the consumer. In one specific embodiment, the Web server associated with the airline (or the airline itself) keeps track of the consumer's account and deducts the mileage. In this instance, an authentication application is used to validate the presence of the card or to obtain access to the Web server site.

Detailed Description Text (95):
Next, similar to step 606, web server 208' sends a page of information to client terminal 204. When claiming benefits, the total cost field is zero and the currency field is a specially assigned value. Keeping total cost field equal to zero causes the system to perform authentication but not to create a payment record. Alternatively, for those user's whose card holds the amount of their points, additional fields may be sent from server 208' to terminal 204 indicating which account to debit and by how many points. The total cost and currency fields may be readily adapted for this purpose.

Detailed Description Text (96):
Next, in a similar fashion to steps 608-612, a draw request message is built, and the draw request is sent to authentication server 206' over link 236'. Similar to step 614, the authentication server now processes the draw request in conjunction with security card 218 (for example) and sends back a "debit" command and a security card signature to authentication server 206'. As total cost is zero, the "draw amount" state reached by security card 218 is also zero. In the alternative

embodiment in which stored-value card 5 stores points for a particular program, total cost may be a value and a "draw amount" state may be reached indicating a number of points to be deducted from card 5.

Detailed Description Text (97):
Next, similar to steps 616-618, authentication server 206' sends the debit command and security card signature to client terminal 204 and this information is processed by card 5. Even though a monetary value is not being debited, card 5 performs processing such as incrementing a counter indicating number of transactions and generating a stored-value card signature. In the alternative embodiment in which points are stored on card 5, the points needed to redeem the benefit chosen by the user from web server 208' may be debited from the appropriate account in this step.

Detailed Description Text (98):
Steps 620 through 638 are performed in a similar manner as in FIGS. 11B and 11C, except that in this case a monetary transaction is not being verified, but rather card 5 is being authenticated to allow the user to complete his access to services or benefits. In step 626 in particular, the signature of card 5 is verified by security card 218. In this embodiment, security card 218 would send an "authentication OK" message rather than the "confirmation" message of step 628. Web server 208' then debits the appropriate number of points from the user's account or allows access to a privileged service for the benefit requested. In the alternative embodiment in which points are stored on card 5, the "authentication OK" message serves not only as an authentication of card 5, but also confirmation that the correct number of points have been debited from card 5 for the appropriate program. Next, similar to step 640, web server 208' releases the benefit requested by the user (such as airline tickets, prizes, discounts, etc.) and the benefit is arranged to be delivered to the user.

Detailed Description Text (105):
Briefly, system 850 operates as follows. A consumer accesses bank server 860 via client terminal 204. Assuming that card 5 is not overloaded and that the user's account with the bank has sufficient funds, the user is able to download value via bank server 860 on to his stored-value card 5. Client terminal 204 communicates with load server 862 to receive authorization for the load and for higher security. Card 5 may then be used to make purchases over the Internet as described earlier in the application or may be used for purchases elsewhere. Once the bank has downloaded value to card 5, a corresponding amount of funds is transferred from the bank to card issuer 108.

Detailed Description Text (111):
To determine the load value, the bank server requests that the user enter the amount to load to the card. Assuming that the user's account is adequate, the bank server requests the user's account be debited in step 875 by the load value. Advantageously, the debit request from the bank server can use the existing ATM and accounting systems of the bank to debit the user's account. From the bank's point of view, value is being transferred from the user's account much in the same way that value would be transferred to a user in the form of cash at an ATM. In this situation, though, the value is not being dispensed as cash at an ATM, but is being sent over the Internet to a stored-value card.

Detailed Description Text (113):
The client terminal emulates a variety of host security module 864 commands to receive responses from these commands from the stored-value card. The stored-value card and the security module are physically separated from one another; communication takes place over the Internet. In the interest of speed and reliability, it is advantageous to have only the traditional authentication, response, and confirmation messages exchanged.

Detailed Description Text (128):
Returning now to a more detailed discussion of step 879, FIG. 11D describes a technique for processing a load request message in conjunction with a security module. Once the load request message is received by the load server, the load server parses it into the appropriate elements and passes a request to the security module as will be explained below. Alternatively, the load server can build a network message and switch the request to a remote authentication server. Or, a smart terminal could parse the message and pass responses to the security module.

Previous Doc Next Doc Go to Doc#